Amendments to the Specification:

Please replace the paragraph beginning at page 2, line 1 with the following amended paragraph:

A mobile application, sometimes also called a mobile app or a mobile agent, is a currently executing computer software application/program, or part of a currently executing computer program that can physically move from one computer to another (between hosts) while it is being executed: A mobile application's software may or may not have been previously installed on a particular ~~computers~~ computer prior to the arrival of the mobile application. The mobile applications are said to jump from one computer to another computer and the process of jumping from one computer to another computer is also referred to as a jump.

Please replace the paragraph beginning at page 3, line 16 with the following amended paragraph:

A Trusted Computing Base (TCB) is the collection of computers, computer peripherals, and communication networks which must perform all requested operations properly, and must not perform extraneous operations, and are trusted to do so, in order to properly complete whatever computations are required. A host outside of the TCB can perform nefarious tasks on the mobile application. This nefarious behavior cannot be controlled, and it cannot be detected. Therefore, once a mobile application has visited an untrusted host, it could be altered in an undesirable way, and therefore is a security hazard. In addition, the mobile application that visited the untrusted host can no longer be trusted to execute within the TCB. All of these security problems with mobile ~~application~~ applications need to be overcome before mobile applications become more accepted as an alternative to traditional computing systems. Thus, it is desirable to provide a mobile application security system and method that overcomes the above problems and limitations with conventional mobile application systems and it is to this end that the present invention is directed so that mobile applications may be used in most financial, commercial, and military computer systems.

Please replace the paragraph beginning at page 10, line 13 with the following amended paragraph:

Figure 3 is a diagram illustrating the movement of a mobile application 40 in a conventional peer-to-peer mobile application system 42. In this example, the system 42 may include one or more host computers, such as Host1, Host2, Host3, Host4 and Host5, that execute the mobile application ~~are~~ at different times as the mobile application jumps between the hosts as is well known. As shown in Figure 3, the mobile application 40 may jump directly from one host to another host such that there is never a central repository for information about the mobile application. Thus, a noted problem with the mobile application 40 from Host 1 may never be known by the other Hosts. In addition, any of the Hosts in the system 42 may sabotage or alter the mobile application 40 to perform some nefarious act, such as placing a virus into the mobile application 40. It is desirable to provide a system wherein the hosts and the mobile application are protected from attacks and the invention solves these problems as will now be described.

Please replace the paragraph beginning at page 11, line 11 with the following amended paragraph:

The server 52 may include a CPU 58 and a memory 60 along with a persistent storage device (not shown) for permanently storing one or more software applications or modules that may be executed by the CPU by loading the software applications or modules into the memory. The server may also include a database 62 that stores one or more mobile applications along with information about the mobile applications as described below. As shown, the memory of the server has a mobile application controller module 64 stored in it that, when executed by the CPU, ~~control~~ controls the security of the mobile applications and hosts as described below. In a preferred embodiment, the mobile application controller 64 may be one or more software ~~application~~ applications or modules, but the controller may also be implemented using hardware.

Please replace the paragraph beginning at page 11, line 20 with the following amended paragraph:

In a preferred embodiment, the mobile application controller 64 may include security

~~software~~ module 66 and a communications ~~software~~ module 68. The combination of the ~~software~~ modules may solve the problems with typical mobile application systems so that: 1) A hostile host cannot send code with undesirable behavior to another host; 2) A mobile application can ~~cant~~ be protected from a hostile host; and 3) A mobile application can be securely sent to or received from a host outside of a group of trusted computers, known as the Trusted Computing Base (TCB) without fear of hostile activity. The way in which the security system in accordance with the invention overcomes these problems will now be described.

Please replace the paragraph beginning at page 12, line 7 with the following amended paragraph:

Figure 5 is a diagram illustrating the operation of the mobile application security system 50 of Figure 4. In particular, the security system 50 in accordance with the invention uses a client/server based security model as opposed to the typical peer-to-peer arrangement as shown in Figure 3. Thus, using the security system 50 in accordance with the invention, there is centralized server 52 which is not a host for the mobile applications, but acts as a server for the participating hosts (Host1, Host2, Host3, Host4 and Host 5 in this example) that are the clients. Thus, in accordance with the invention, each of these clients (Hosts) communicates with only the server 52 and never directly with each other. Thus, as shown in Figure 5, the mobile application 40 must pass through the server 52 on each jump between the hosts.

Please replace the paragraph beginning at page 12, line 16 with the following amended paragraph:

Figure 6 is a diagram illustrating more details of the mobile application security system 50 shown in Figure 5. In particular, the client/server architecture of the security system 50 in accordance with the invention ensures that the server 52 tracks all of the mobile applications in the system and all of the jumps of all of the mobile applications. The server 52 may also perform security procedures on the mobile applications while they are in transit. Thus, for example, a security check 70 may be performed by the security module 66 of the server 52 each time a mobile application jumps from one host to another host as shown in Figure 6.

Please replace the paragraph beginning at page 17, line 10 with the following amended paragraph:

Figure 8 is a diagram illustrating a first example of a second embodiment of the mobile application security system 50 for preventing hostile hosts from transmitting code to other hosts in accordance with the invention. In particular, the mobile application 40 is created by Host1 and then later dispatched to another host to continue the execution of the mobile application 40. In this example, Host1 is untrusted in that the server 52 does not know whether or not to trust the host when interacting with the mobile application 40. Therefore, the mobile application 40 dispatched from Host1 is sent to the server 52 in accordance with the invention and the server 52 may perform several security measures. For example, it may strip any code from the mobile application 40 and store an (empty) copy of the mobile application code in the database 62. The server 52 may alternatively check the code to ensure that it is safe and forward only safe code to the next host. The server 52 may then forward the mobile application 40 onto the next host, Host2 in this example. The mobile application 40 may then be received by and executed by Host2. When the mobile application 40 requires code for execution, the tested version of the code may be supplied to Host2 by the server 52 thus ensuring that the untrusted host cannot spread a virus, for example, using the mobile application 40. Now, the dispatch of a mobile application from a trusted host to another host will be described.

Please replace the paragraph beginning at page 18, line 5 with the following amended paragraph:

Figure 9 is a diagram illustrating a second example of a second embodiment of the mobile application security system 50 for preventing hostile hosts from transmitting code to other hosts in accordance with the invention. In particular, the mobile application 40 is created by Host1 and then later dispatched to another host to continue the execution of the mobile application 40. In this example, Host1 is trusted in that the server 52 knows that the particular host is trusted and therefore does not need to strip the code from the mobile application 40 and test it as described above. Therefore, the mobile application 40 dispatched from Host1 is sent to the server 52 in accordance with the invention and the server 52 may store a copy of the mobile

application code in the database 62. The server 52 may then forward the mobile application 40 onto the next host, Host2 in this example. The mobile application 40 may then be received by and executed by Host2. When the mobile application 40 requires the code for execution, the known safe version of the code may be supplied to Host2 by the server 52 or, since the originating host is trusted, the code may be provided by the originating host. Now, the subsequent dispatch of a mobile application from an untrusted host will be described.

Please replace the paragraph beginning at page 18, line 19 with the following amended paragraph:

Figure 10 is a diagram illustrating a third example of a second embodiment of the mobile application security system 50 for preventing hostile hosts from transmitting code to other hosts in accordance with the invention. In particular, the mobile application 40 is received from another host by an untrusted host (Host n) and then later dispatched to another host to continue the execution of the mobile application 40. In this example, Host n is untrusted in that the server 52 does not know whether the particular host may perform nefarious acts on the mobile application or using the mobile application. Therefore, the mobile application 40 dispatched from Host n is sent to the server 52 in accordance with the invention and the server 52 may perform several security measures. For example, the server 52 may receive the code of the mobile application 40 and compare the current code to a previously stored version of the code to ensure that the newly received code is the same as the previous code. The server 52 may then forward the mobile application 40 onto the next host, Host n+1 in this example. The mobile application 40 may then be received by and executed by Host n+1. When the mobile application 40 requires code for execution, the known safe version of the code may be supplied to Host n+1 by the server 52 or, if the originating host is trusted, the code may be provided by the originating host. Now, the subsequent dispatch of a mobile application from a trusted host will be described.

Please replace the paragraph beginning at page 19, line 14 with the following amended paragraph:

Figure 11 is a diagram illustrating a fourth example of a second embodiment of the

mobile application security system 50 for preventing hostile hosts from transmitting code to other hosts in accordance with the invention. In particular, the mobile application 40 is received from another host by a trusted host (Host n) and then later dispatched to another host to continue the execution of the mobile application 40. In this example, Host n is trusted in that the server 52 knows that the particular host will not perform nefarious acts using the mobile application 40. Therefore, the mobile application 40 dispatched from Host n is sent to the server 52 in accordance with the invention and the server 52 may perform several security measures. For example, the server 52 may receive the code of the mobile application 40 and store a copy of it in the database 62. No comparison is necessary since the host is trusted. The server 52 may then forward the mobile application 40 onto the next host, Host n+1 in this example. The mobile application 40 may then be received by and executed by Host n+1. When the mobile application 40 requires the code for execution, the known safe version of the code may be supplied to Host n+1 by the server 52 or, if the originating host is trusted, the code may be provided by the originating host. Now, a third embodiment of the mobile application security system will be described.

Please replace the paragraph beginning at page 22, line 14 with the following amended paragraph:

Figure 12 is a diagram illustrating a third embodiment of the mobile application security system 50 for detecting unwanted changes to the state of a mobile application in accordance with the invention. In general, the server 52 may compare the state of the mobile application on the previous jump with the state of the mobile application on the current jump. This allows the server to detect the unwanted changes in the state of the mobile application. In more detail, a host, Host1 in this example, may create a mobile application 40 that is then dispatched to other hosts for further execution. When the mobile application 40 is dispatched, it is sent to the server 52 which may save a copy of the mobile application's state (e.g., in storage 62). The server 52 may then forward the mobile application 40 to the next host, Host2 in this example. Host2 may receive the mobile application 40, execute it and then forward it onto the next host. The server 52 may receive the mobile application 40 from the next host and compare the state of the mobile

application 40 received from the next host to the state of the mobile application 40 saved in the database to determine if changes have occurred. If the comparison does not detect any unwanted changes with the mobile application 40, the server 52 may forward the mobile application 40 onto the next host. Thus, in this embodiment, a host that executes the mobile application 40 is unable to insert changes into the mobile application's state since those changes will be identified by the server 52 when the comparison step is executed by the server 52. Now, a fourth embodiment of the mobile application security system will be described.

Please replace the paragraph beginning at page 23, line 16 with the following amended paragraph:

Figure 13 is a diagram illustrating a first example of a fourth embodiment of the mobile application security system 50 for detecting unwanted changes in the itinerary of the mobile application in accordance with the invention. In general, on each jump of the mobile application, the server may determine the host from which the mobile application was dispatched and the hosts to which the mobile application is dispatched. In particular, this permits the server 52 to enforce the itinerary (e.g., the hosts where the mobile application is going to be executed) of the mobile application. In more detail, a first host (Host1) may create a mobile application 40 and then may dispatch the mobile application 40 to another host through the server 52 in accordance with the invention. When the server 52 receives the mobile application 40, the server 52 may store a copy of the itinerary of the mobile application 40 in the database 62. The server 52 may then forward the mobile application 40 to the next host (Host2) according to the itinerary. Now, another example of the embodiment for detecting changes in the itinerary will be described.

Please replace the paragraph beginning at page 24, line 7 with the following amended paragraph:

Figure 14 is a diagram illustrating a second example of a fourth embodiment of the mobile application security system 50 for detecting unwanted changes in the itinerary of the mobile application in accordance with the invention wherein the itinerary of a mobile application is already stored in the server. In more detail, a first host (Host n) may dispatch a mobile

application 40 to another host through the server 52 in accordance with the invention. When the server 52 receives the mobile application 40, the server 52 may compare the current itinerary of the mobile application 40 to a stored copy of the itinerary to ensure they match each other. If the itineraries match, then the server 52 may forward the mobile application 40 onto the next host (Host n+1) that receives the mobile application 40 and executes it. Now, another example of the embodiment for detecting changes in the itinerary will be described.

Please replace the paragraph beginning at page 24, line 17 with the following amended paragraph:

Figure 15 is a diagram illustrating a third example of a fourth embodiment of the mobile application security system 50 for detecting unwanted changes in the itinerary of the mobile application in accordance with the invention wherein the itinerary may be changed. In more detail, a first host (Host n) which has received a mobile application 40 from another host may dispatch the mobile application 40. The mobile application 40 then passes through the server 52 in accordance with the invention. When the server 52 receives the mobile application 40 in accordance with the invention, it may ensure that the historical portion of the itinerary is accurate by comparing the previously saved itinerary with the new itinerary. If the historical portion of the itinerary is accurate, the server 52 forwards the mobile application 40 to the next host (Host n+1). Now, a fifth embodiment of the mobile application security system will be described.

Please replace the paragraph beginning at page 25, line 6 with the following amended paragraph:

Figure 16 is a diagram illustrating a first example of a fifth embodiment of the mobile application security system 50 for preventing untrusted hosts from launching a mobile application in accordance with the invention. In general, on each jump of the mobile application, the server may determine if the mobile application has previously been in the system. For example, if the host from which the mobile application is sent is an untrusted host, the server may prevent the mobile application from being forwarded to the next host. In more detail, as shown in Figure 16, a first host (Host1) may create a mobile application 40 and then later

dispatch it to another host. In accordance with the invention, the dispatched mobile application 40 first is sent to the server 52. The server 52 may determine that the mobile application 40 is new and therefore further investigation is necessary. If the server 52 then determines that the particular host is allowed (e.g., is trusted to) to launch mobile applications, the server 52 may forward the mobile application 40 to the next host (Host2) so that Host2 receives the mobile application 40.

Please replace the paragraph beginning at page 25, line 18 with the following amended paragraph:

Figure 17 is a diagram illustrating a second example of a fifth embodiment of the mobile application security system 50 for preventing untrusted hosts from launching a mobile application in accordance with the invention. In particular, an untrusted host (Host1) may create a new mobile application 40 that is then later dispatched. The mobile application 40 is then ~~sent~~ dispatched to the server 52 first in accordance with the invention. The server 52 determines that the host dispatching the mobile application 40 is untrusted so that the server 52 does not forward the mobile application 40 to the next host.

Please replace the paragraph beginning at page 26, line 4 with the following amended paragraph:

Figure 18 is a diagram illustrating a third example of a fifth embodiment of the mobile application security system 50 for preventing untrusted hosts from launching a mobile application in accordance with the invention wherein a subsequent dispatch of the mobile application occurs. In particular, a host (Host n) attempts to dispatch a mobile application 40 to another host which must pass through the server 52 in accordance with the invention. When the mobile application 40 is received by the server 52, the server 52 may determine that the mobile application 40 is not new (e.g., the server 52 knows about the mobile application 40 and knows that it is safe) and forwards the mobile application 40 to the next host (Host n+1). Now, a summary of how the above procedures raise the security level of a mobile application environment will be described.